ANCHOR TOOLKIT

Project Participants

Srilekha S. Mudumbai

Abdelilah Essiari

Dilpreet Singh

Bill Johnston

Lawrence Berkeley National Laboratory

Information and Computing Sciences Division (ICSD)

OVERVIEW

- *Agent*
- *Mobile Agent*
- *Why Mobile Agents*
- *Mobile Agent Security*
- *Applications in a Collaborative Framework*
- *Partnerships*
- *Java Limitations*
- *Anchor Toolkit*

Agent

- Software Agents
- Piece of loadable code
- Programs that assist people and act on their behalf
- Continuous and Autonomous by nature
- Adaptive
- Cooperative
- Collaborative
- Smart

Mobile Agent

- Dynamic Behavior
- Ability to Migrate from one host to another
  (can be transported from one machine to another in the middle of their execution)
- Creation of new contexts during migration is necessary so that the run-time environment is transparent for the agents.

Why Mobile Agents

- Reduced Network Load
  (advantage of the code on the same host or
  the network as the target objects)
- Overcome network latency
- Encapsulate protocol
- Execute asynchronously and autonomously
- Adapt dynamically
- Naturally heterogeneous (JAVA)
- Robust and fault-tolerant
- Flexible Remote Execution

Mobile Agent Security

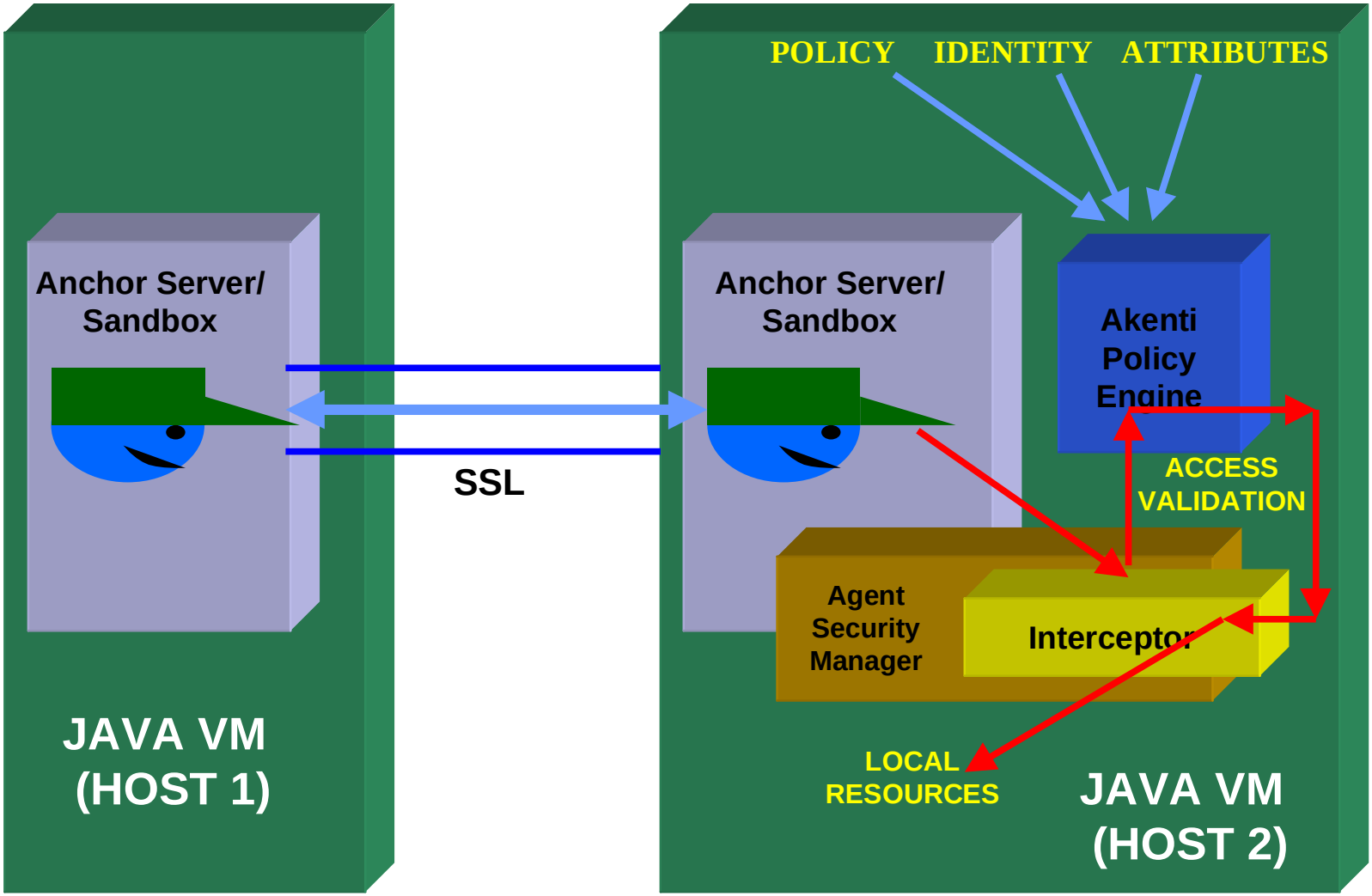
INTERCEPTOR:

- **The whole system has been implemented in Java as it offers several features suitable for a mobile agent system. Java's security model provides an intercepting mechanism for resource level calls.**

- **Java allows you to provide your own security mechanism that can override its default. These interceptors are able to interpret any actions carried out by mobile agents as in reading, writing, and executing files, opening sockets etc.**

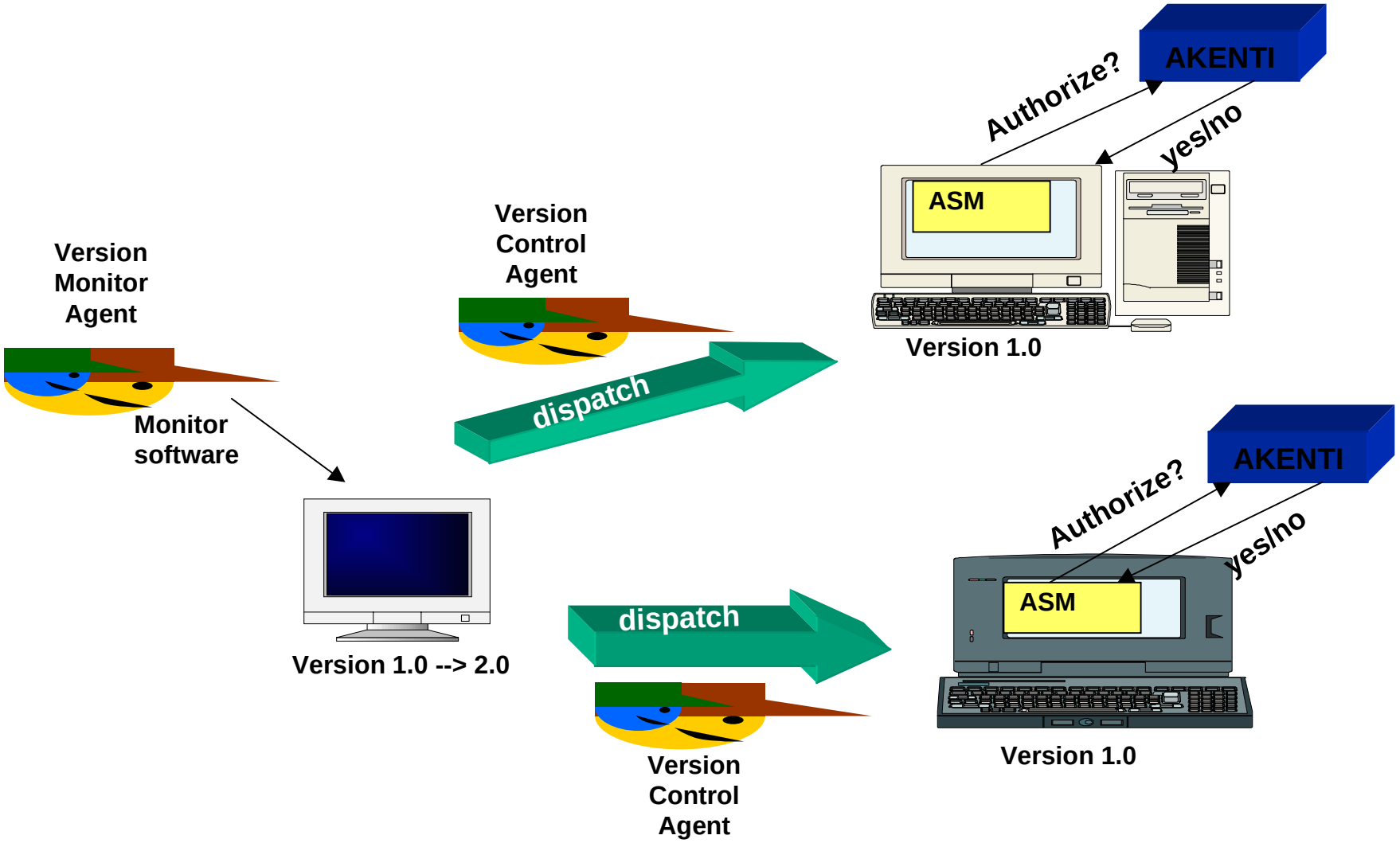
- **Java also provides portability of code across heterogeneous platforms and object serialization to encode objects into a stream of bytes that can later be reconstructed from the stream.**

Mobile Agent Security



Applications in a Collaborative Framework

Software Upgrades over the Network



Collaborative Applications

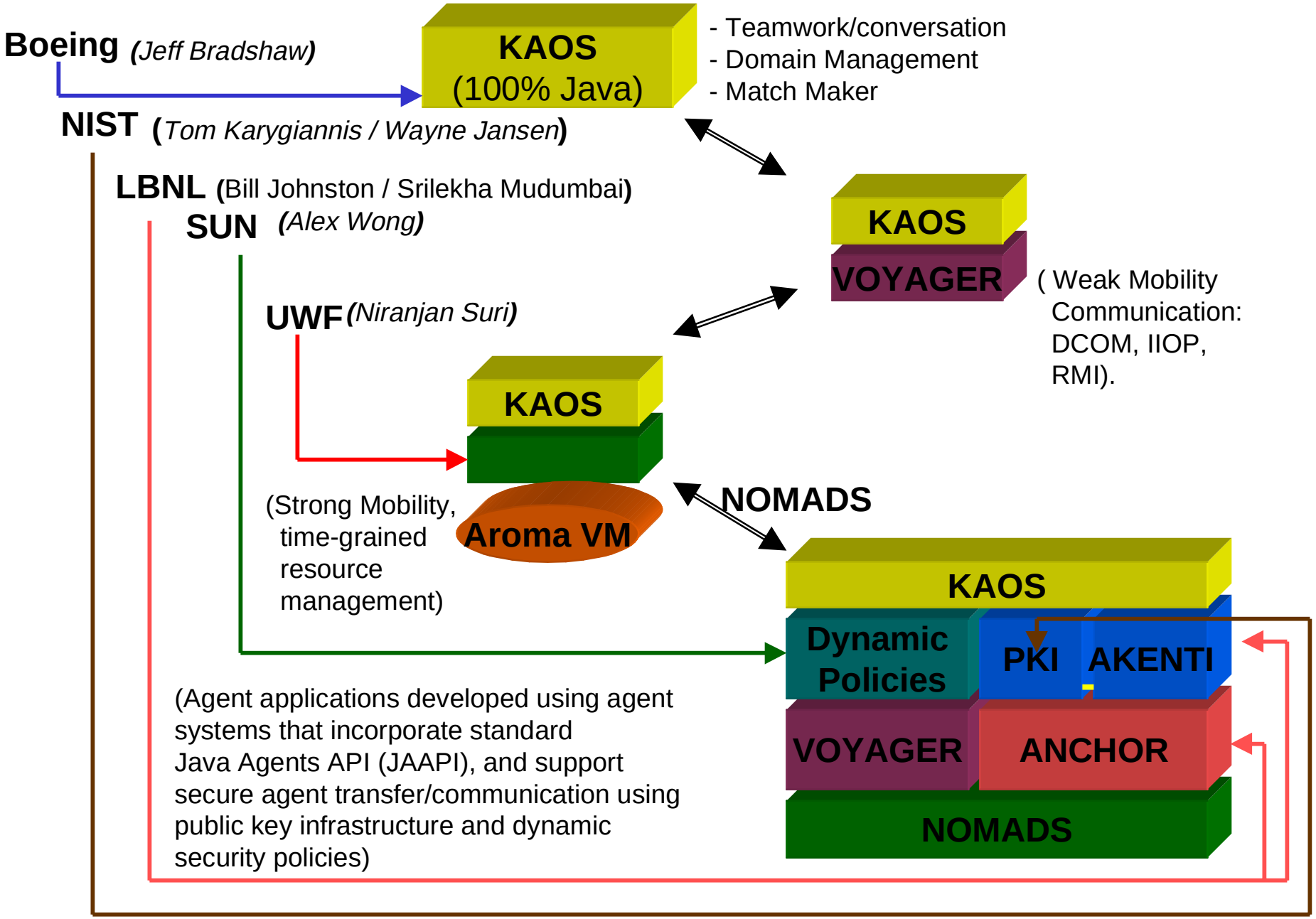
- Secure data sharing and access control for users in a collaboration.

- Control of one or more instruments through mobile agents. Mobile code can be sent to these instruments so that the real-time operations can be obtained. Code level control can be obtained according to the classification of the instruments. Operation level control can be obtained depending on the users accessing the instrument.

Java Limitations

- Inadequate support for resource control (no way to limit the resources consumed by an agent)
- No support for preservation and resumption of the execution state
- Static access policy files must reside on each host machine

Partnerships



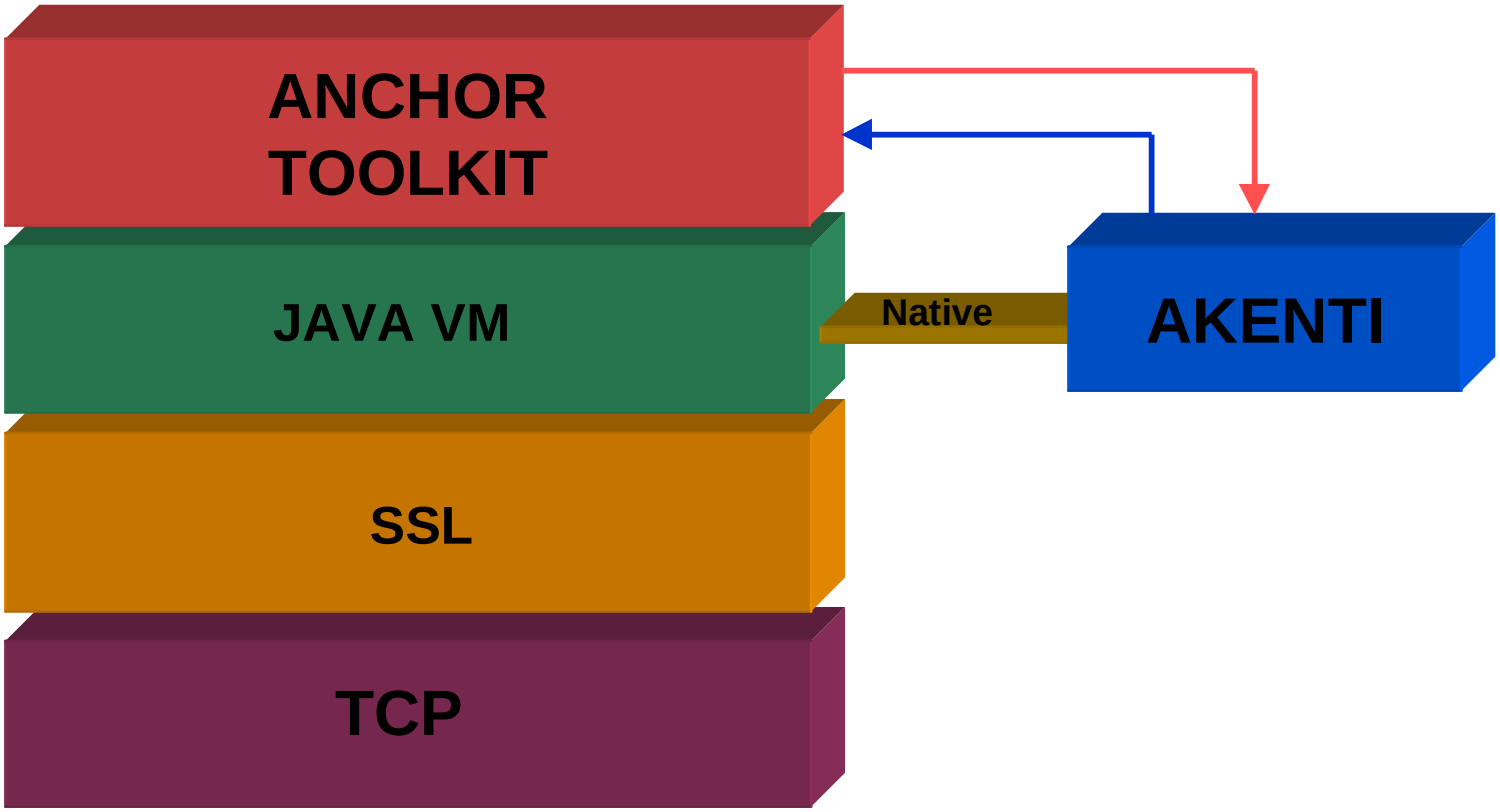
Partnerships

INTEGRATION EFFORT

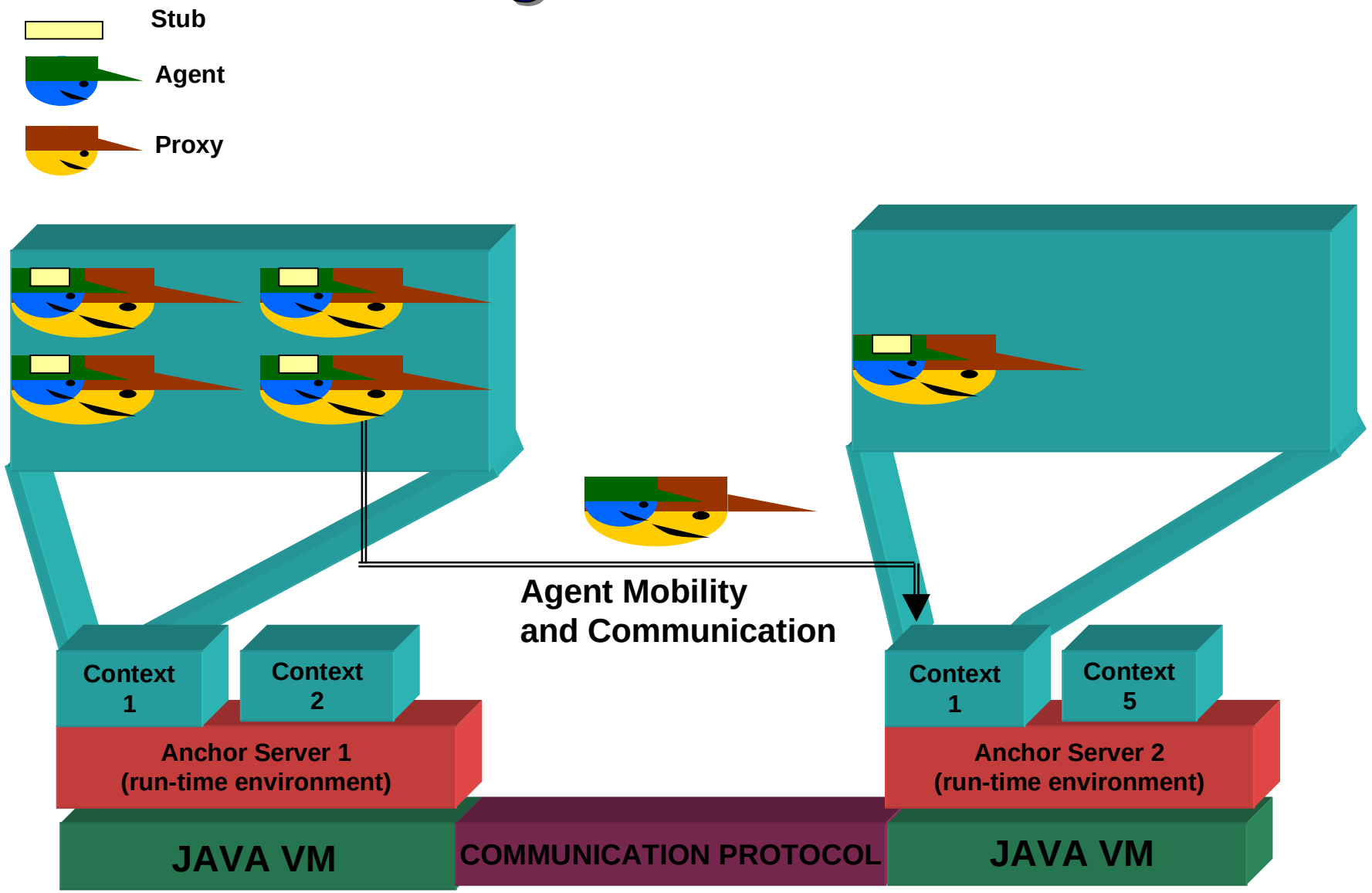
The objective of the integration effort is to demonstrate the usefulness of attribute certificates for mobile agents in controlling access and use-conditions of distributed computing resources.

- NOMADS gives the "anytime" mobility and fine grained resource control
- Dynamic Security Policy - Sun
- KAoS - Boeing
- Access Control : Akenti/Anchor Toolkit - LBNL

Anchor Toolkit



Agent Model


Stub
Agent
Proxy
Agent Mobility
and Communication
Context
1
Context
2
Anchor Server 1
(run-time environment)
JAVA VM
COMMUNICATION PROTOCOL
Context
1
Context
5
Anchor Server 2
(run-time environment)
JAVA VM